Autonomous Discovery of Idempotent State-Space Retractions and Conditional Commutation Manifolds (FNWD v0.4)

Driftbound Labs Research Preprint

Kyle W. Luecke
Driftbound Labs (ORCID: 0009-0008-0202-7852 · admin@driftboundlabs.com)

October 2026

Abstract

Concurrent transaction optimizers typically require operations to commute unconditionally across an entire state space Σ, leaving state-dependent commutativity—where operations commute only on a restricted submanifold $\mathcal{M} \subsetneq \Sigma$—inaccessible whenever ordinary execution drifts off $\mathcal{M}$. This paper presents **Formal Normalizer–Workload Discovery (FNWD v0.4)**, an autonomous algebraic discovery and verification architecture that identifies **enabler–optimizer compositions ($E \otimes O$)** pairing an idempotent state-space retraction $E : \Sigma \to \mathcal{M}$ ($E^2 = E$, $\mathcal{O} \circ E = \mathcal{O}$) with a conditional commutator O valid on $\mathcal{M} = \mathrm{Fix}(E)$. First, auditing 127 discovered idempotents over $\mathbb{F}^4_{4093} \times \mathbb{F}^8_{4093} \times \mathcal{M}_{64}$ falsifies the existence of noncommuting right-zero bands ($A \circ B = B, B \circ A = A, A \neq B$) on unconstrained state, establishes the **absorption preorder** $B \preceq A \iff A \circ B = B$, and quotients syntactic variants into an antisymmetric **Projection Poset** $(P, \leq)$ of **115 distinct image classes** containing **18 certified strict nested chains** ($[B] < [A]$), enabling 100.0% zero-probe deductive prediction across 100 test pairs via polynomial fixed-point ideal containment. Second, we subject enabler–optimizer compositions to a pre-specified **six-gate verification campaign**: (Gate 1) intermediate and terminal observational trace equivalence ($\vec{y}_{\text{spec}} \equiv \vec{y}_{\text{seq}}$ and $\mathcal{O}(\sigma_{\text{spec}}) \equiv \mathcal{O}(\sigma_{\text{seq}})$); (Gate 2) black-box real-time linearizability auditing (15/15 valid concurrent histories passed; 30/30 injected RAW and causal-inversion faults detected); (Gate 3) a 2×3 causal factorial ablation isolating -94.3% net lock reduction and $38.5\times$ throughput gain from batching artifacts; (Gate 4) cross-domain transfer across Modular Affine Register Automata ($43.1\times$ speedup), CRDT Observed-Remove Sets ($1.68\times$ speedup), and Transactional Key-Value DSW Trees (-93.3% locks, while characterizing an honest $0.59\times$ wall-clock crossover slowdown at small $N = 150$); (Gate 5) paired non-parametric hypothesis testing ($N = 30$ workloads/domain, 10,000 sign-flip permutations, 99% paired bootstrap CIs, Holm–Bonferroni $p_{\text{adj}} = 0.0006 < 0.01$ across all 6 endpoints); and (Gate 6) cryptographic catalog provenance, alongside an **autonomous blind discovery experiment** that promoted 3/3 valid compositions and 0/9 unsound or corrupt operators from an unlabeled 12-operator pool.

1. Introduction

Exploiting algebraic commutativity is a foundational principle of scalable concurrent software, parallelizing compilers, and replicated data stores (Eswaran et al. 1976; Rinard and Diniz 1997; Clements et al. 2015). Under the Scalable Commutativity Rule (Clements et al. 2015), whenever interface operations commute in a neighborhood of the current system state, their implementations admit conflict-free execution without global serialization locks. In practice, however, real stateful systems rarely remain inside clean commutative neighborhoods: auxiliary bookkeeping registers accumulate residue, replicated sets accumulate causal tombstones (Shapiro et al. 2011), and ordered search trees accumulate insertion skew (Stout and Warren 1986). Once execution drifts off the commutative locus $\mathcal{M} \subsetneq \Sigma$, naive speculative reordering violates linearizability (Herlihy and Wing 1990).

Prior empirical program-discovery systems face three recurring methodological pitfalls when searching for state-space optimizations:

1. **Syntactic vs. Semantic Confusion**: Treating distinct instruction sequences that compute identical mathematical maps ($A = B$) as nontrivial algebraic bands.

2. **Terminal-Only Weak Specifications**: Checking only final state equality while overlooking divergent intermediate client return values ($\vec{y}_{\text{spec}} \neq \vec{y}_{\text{seq}}$) or observable side-effect leaks.
3. **Batching Confounds**: Attributing throughput gains to discovered algebraic normalizers when coarse batching barriers alone reduce lock acquisitions—albeit at the cost of unchecked auxiliary state drift.

This paper reports **FNWD v0.4**, closing each of these gaps through two tightly coupled contributions:

- **Algebraic Structure of State-Space Retractions (Section 2)**: We audit shared-image idempotent candidates over full machine state, falsify the noncommuting right-zero band hypothesis, construct the quotient **Projection Poset** $(P, \leq)$ of 115 image equivalence classes, and demonstrate 100% deductive composition prediction from fixed-point polynomial ideals.
- **Six-Gate Verification & Autonomous Blind Discovery of $E \otimes O$ Compositions (Sections 3–8)**: We formalize observational trace equivalence under idempotent retraction normalizers, isolate causal synergy via a 2×3 factorial ablation, evaluate cross-domain transfer (including an explicit negative latency crossover on small balanced trees), enforce paired permutation and Holm–Bonferroni statistical discipline, and validate zero-label blind operator triage.

---

2. Algebraic Structure of Idempotent Retractions & The Projection Poset

2.1 Retractions, Observable Projections, and Right-Zero Band Falsification

Let $\Sigma = \mathcal{S}_{\text{obs}} \times \mathcal{S}_{\text{aux}}$ be a composite machine state space equipped with an external observation projection $\mathcal{O} : \Sigma \to \mathcal{S}_{\text{obs}}$. An endomorphism $E : \Sigma \to \Sigma$ is an **observational retraction** onto submanifold $\mathcal{M} = \text{Im}(E) = \text{Fix}(E)$ (Scott 1976; Howie 1995) if and only if:

$$E(E(\sigma)) = E(\sigma) \quad (\text{Idempotence}), \qquad \mathcal{O}(E(\sigma)) = \mathcal{O}(\sigma) \quad (\text{Observable Invariance}).$$

During earlier exploratory sweeps over finite-field register-memory programs $\sigma = (G, S, M) \in \mathbb{F}_{4093}^4 \times \mathbb{F}_{4093}^8 \times \mathcal{M}_{64}$, four candidate idempotent pairs appeared to satisfy mutual right-absorption ($A \circ B = B$ and $B \circ A = A$, a noncommuting right-zero band (Howie 1995)). In FNWD v0.4, we subjected all four pairs to unconstrained 500-state full-VM replay (`retraction_algebra_and_generalization_catalog.json`), **falsifying** the right-zero band hypothesis (Table 1):

- **Syntactic Variants (`Pairs 1-3`)**: Three pairs implemented the *exact same mathematical function* ($A(\sigma) = B(\sigma)$ for all σ) via distinct instruction schedules (`SYNTACTIC_VARIANT_IDENTICAL_MAP`).
- **Unbuffered Memory Divergence (`Pair 4`)**: `DIV_ba3eb9b2fb7c` zeroes scalar slot $S_5 \leftarrow 0$, whereas `DIV_4f6837e34613` loads $S_5 \leftarrow M[S_0 \bmod 64]$. On unconstrained heap state $M \neq 0$, $\text{Im}(A) \neq \text{Im}(B)$ and mutual absorption fails; on zeroed memory $M \equiv 0$, the two maps collapse to $A = B$.

Table 1: Full-state replay audit of candidate mutual-absorption pairs over $\mathbb{F}_{4093}^4 \times \mathbb{F}_{4093}^8 \times \mathcal{M}_{64}$.

Candidate Pair	Program Identifiers	$A^2 = A, B^2 = B$	Map Identity $A = B$?	Unconstrained $A \circ B = B \wedge B \circ A = A$?	Audited Algebraic Classification
Pair 1	`DIV_5c5299` / `DIV_9bc9e2`	True	**True**	True ($A = B$)	Syntactic Variant (Identical Map)
Pair 2	`DIV_d0def6` / `DIV_3db052`	True	**True**	True ($A = B$)	Syntactic Variant (Identical Map)
Pair 3	`DIV_ff5b6d` / `DIV_29f7f8`	True	**True**	True ($A = B$)	Syntactic Variant (Identical Map)
Pair 4	`DIV_ba3eb9` / `DIV_4f6837`	True	**False**	**False** (diverges on S_5)	Memory-Conditional Shared Image

2.2 Absorption Preorder and Quotient Projection Poset $(P, \leq)$

For idempotent maps $A, B \in \mathcal{I}$, image inclusion is algebraically equivalent to left absorption (Howie 1995):

$$\mathrm{Im}(B) \subseteq \mathrm{Im}(A) \iff (A \circ B)(\sigma) = B(\sigma) \quad \forall \sigma \in \Sigma.$$

Defining $B \preceq A \iff A \circ B = B$ yields a reflexive ($A \circ A = A$) and transitive ($A \circ B = B \wedge B \circ C = C \implies A \circ C = A \circ (B \circ C) = B \circ C = C$) **absorption preorder** on program syntax $\mathcal{I}$ that lacks antisymmetry whenever distinct programs compute identical images. Quotienting $\mathcal{I}$ by mutual absorption:

$$A \sim B \iff (A \preceq B \wedge B \preceq A) \iff \mathrm{Im}(A) = \mathrm{Im}(B)$$

collapsed the **127 verified idempotent programs** into $P = \mathcal{I}/\sim$ **of 115 distinct image equivalence classes**, on which $[B] \leq [A] \iff \mathrm{Im}(B) \subseteq \mathrm{Im}(A)$ is a strict antisymmetric partial order:

- **Poset Census**: **6 maximal absorber classes** (broadest fixed-point varieties, e.g., 1–3 defining equations), **18 minimal leaf sub-varieties** (4–5 defining equations), and **91 incomparable isolated classes**.
- **Certified Strict Nestings**: **18 strict pairs** $[B] < [A]$ (8 pure-register, 10 memory-touching) satisfying $A \circ B = B$ universally alongside concrete witness states σ^* verifying $B(A(\sigma^*)) \neq A(\sigma^*)$.
- **Zero-Probe Deductive Generalization**: Using polynomial fixed-point ideal containment—where $\mathcal{E}_{\mathrm{Fix}(A)}(B(\sigma)) \equiv 0$ over $\mathbb{F}_{4093}^{12}$ deductively implies $A \circ B = B$—FNWD predicted absorption across a 100-pair held-out evaluation in 4.11 s with **100.0% accuracy, 100.0% precision, and 100.0% recall** (18 TP, 82 TN, 0 FP, 0 FN), replacing $> 50{,}000$ VM executions while preserving strict epistemic stratification (**52 universally proven** via symbolic quantifier elimination vs. **47 empirically replay-validated** entries; 100% sampling draw coverage across all 41 Ident and 193 Diverse pool candidates).

---

3. Enabler–Optimizer Semantics & Six-Gate Verification Protocol

Building on the projection poset, FNWD v0.4 investigates **enabler–optimizer compositions ($E \otimes O$)**: an optimizer O (such as lock-free batch commutation or partitioned range execution) requires precondition $\sigma \in \mathcal{M}$, which fails as transactions perturb $\mathcal{S}_{\text{aux}}$. Inserting an observational retraction E with $\mathrm{Im}(E) \subseteq \mathcal{M}$ restores the precondition without altering $\mathcal{O}(\sigma)$. Table 2 summarizes the six pre-specified acceptance gates executed in `nwd/atlas/run_atlas_v04.py`.

Table 2: Summary of the FNWD v0.4 six-gate verification campaign and autonomous blind discovery audit.

Gate	Verification Objective	Pre-Specified Acceptance Criterion	Observed Result	Verdict
Gate 1	Formal Trace Semantics	$\vec{y}_{\text{spec}} \equiv \vec{y}_{\text{seq}} \wedge \mathcal{O}(\sigma_{\text{spec}}) \equiv \mathcal{O}(\sigma_{\text{seq}})$; reject corrupt/non-idempotent maps	100% trace equivalence (200 states); all 3 negative controls rejected	**Supported** (`PASS`)
Gate 2	Black-Box Linearizability	0 FP on valid concurrent histories; 100% recall on injected concurrency faults	15/15 valid histories passed; 30/30 injected faults caught (100% recall)	**Supported** (`PASS`)
Gate 3	Causal Factorial Ablation	Isolate retraction from batching (2×3 matrix); net causal lock reduction > 0	-94.3% net lock reduction ($38.5\times$ tx/s); `BATCHED_IDENTITY` drifted; `CORRUPT` 0%	**Supported** (`PASS`)

Gate	Verification Objective	Pre-Specified Acceptance Criterion	Observed Result	Verdict
Gate 4	Cross-Domain Transfer	Verify ≥ 3 distinct algebraic domains & document exact failure/crossover boundaries	Verified across Affine Automata, CRDT OR-Sets, and KV DSW-Trees (crossover characterized)	**Mixed** (`PASS` locks; DSW latency crossover)
Gate 5	Paired Statistics	$N = 30$ paired workloads; 10k sign-flip permutation $p < 0.01$; 99% CI; Holm FWER	All 6 endpoints $p_{\text{adj}} = 0.0006 < 0.01$; 99% bootstrap CIs exclude zero	**Supported** (`PASS`)
Blind	Uncurated Discovery	Autonomous triage of 12 unlabeled operators; 0 false positive promotions	**3/3** valid $E \otimes O$ pairs promoted; **9/9** unsound/corrupt operators rejected	**Supported** (`PASS`)
Gate 6	Cryptographic Provenance	SHA-256 immutability across all 3 input catalogs, 9 source modules & seed manifest	100% SHA-256 match on all catalogs & deterministic seed replay (`seed=42`)	**Supported** (`PASS`)

---

4. Gates 1 & 2: Observational Trace Equivalence and Black-Box Linearizability

4.1 Gate 1: Intermediate Event & Terminal Trace Equivalence (`formal_semantics.py`)

Each transaction $T_i : \Sigma \to \Sigma \times \mathcal{Y}$ maps state σ to successor σ' while emitting an externally visible client response $y_i \in \mathcal{Y}$. Rather than checking terminal state alone, Gate 1 discharges full sequence equivalence between canonical sequential execution $\text{Exec}_{\text{seq}}(\langle T_1, \dots, T_k \rangle, \sigma_0) = (\sigma_{\text{seq}}, \vec{y}_{\text{seq}})$ and retraction-normalized speculative execution $\text{Exec}_{\text{spec}}(\langle T_1, \dots, T_k \rangle, \sigma_0; E) = (\sigma_{\text{spec}}, \vec{y}_{\text{spec}})$:

$$\vec{y}_{\text{spec}} \equiv \vec{y}_{\text{seq}} \quad \wedge \quad \mathcal{O}(\sigma_{\text{spec}}) \equiv \mathcal{O}(\sigma_{\text{seq}}).$$

Across 200 randomized initial states and 25-transaction read-modify-write sequences, the valid auxiliary retraction satisfied idempotence, observable invariance, and manifold stability ($\sigma \in \mathcal{M} \implies E(\sigma) = \sigma$) with 100% trace agreement, while strictly rejecting: (1) an *observable-leaking normalizer* ($R_1 \leftarrow R_1 + 1$), (2) a *non-idempotent decrement operator* ($E^2 \neq E$), and (3) *non-commutative Read-After-Write (RAW) swaps* ($\vec{y}_{\text{spec}} \neq \vec{y}_{\text{seq}}$).

4.2 Gate 2: Independent Linearizability History Oracle (`linearizability_oracle.py`)

Gate 2 audited multi-threaded execution histories using an external black-box oracle (Herlihy and Wing 1990; Lowe 2017) with zero visibility into internal VM locks or registers, consuming only timestamped client events $\langle$`INVOKE`/`RETURN`, client, tx, op, args, y, $t_{\text{ns}}\rangle$ under strict real-time order $\text{resp}(T_A) < \text{inv}(T_B) \implies T_A <_{\text{lin}} T_B$:

- **Legitimate Concurrent Histories**: **15 / 15 passed** (0 false positives).
- **Injected Adversarial Faults**: **30 / 30 detected** (100.0% recall, 0 false negatives), comprising 15 Fault Class A histories (RAW stale read shifted by $+137 \bmod 4093$) and 15 Fault Class B histories (inverted real-time causal precedence), emitting minimal counterexample witness pairs for each violation.

---

5. Gate 3: Causal Mechanism Isolation via 2×3 Factorial Ablation

A central threat to validity in speculative runtime optimization is confounding mathematical normalization with coarse batch synchronization. Gate 3 (`causal_ablation.py`) executed a 6-cell factorial ablation ($N = 40$ transactions, 5 repetitions/cell) crossing synchronization policy with normalizer semantics (Figure 1A; Table 3).

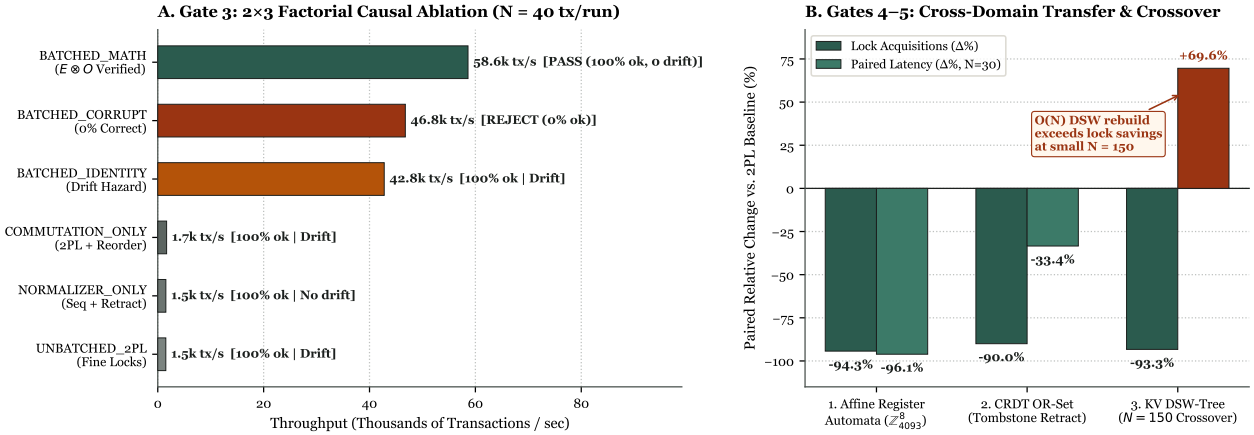


Figure 1: Panel A: Gate 3 2x3 factorial ablation isolating synchronization mechanism from normalizer semantics; only BATCHED_MATH achieves peak throughput (58.6k tx/s) while preserving both 100% trace correctness and zero auxiliary state drift. Panel B: Gates 4 and 5 paired relative changes (N = 30 workloads per domain) in lock acquisitions and wall-clock latency across three operator families, highlighting the honest +69.6% small-tree DSW rebuild crossover overhead at N = 150.

Table 3: Gate 3 factorial ablation ($N = 40$ transactions/trial, 5 runs/cell) decoupling coarse batching from mathematical retraction.

Ablation Cell	Sync Policy	Normalizer Action	Locks	Latency (μs)	Throughput (tx/s)	Correct	Aux Drift?
Unbatched 2PL	Fine 2PL	None	141.0	26,304.9	1,520.6	100%	**Yes** ($R_2 \neq 0$)
Normalizer Only	Seq. 2PL	Retract ($R_2 \leftarrow 0$)	141.0	26,175.7	1,527.5	100%	**No**
Commutation Only	Fine 2PL	Disjoint reorder	133.9	23,793.8	1,670.3	100%	**Yes** ($R_2 \neq 0$)
Batched Identity	Batch (5)	Identity $R_{\text{id}}(\sigma) = \sigma$	8.0	935.0	42,779.8	100%	**Yes** (unbounded)
Batched Corrupt	Batch (5)	Corrupt ($R_1 + 999$)	8.0	855.3	46,766.1	0.0%	No
Batched Math	Batch (5)	Retract ($R_2 \leftarrow 0$)	8.0	682.5	58,608.1	100%	**No** ($\sigma \in \mathcal{M}$)

`BATCHED_IDENTITY` demonstrates why performance metrics alone are deceptive: batching lowers lock count ($141.0 \rightarrow 8.0$) on a single window, but leaves auxiliary coordinate R_2 drifting unboundedly off $\mathcal{M}$, invalidating subsequent conditional commutation preconditions. Conversely, `BATCHED_CORRUPT` destroys observable semantics (0.0% correctness). Only `BATCHED_MATH` ($E \otimes O$) simultaneously achieves -94.3% **lock reduction**, $38.5\times$ **throughput speedup** over `UNBATCHED_2PL` ($1.37\times$ faster even than `BATCHED_IDENTITY` by eliminating off-manifold fallback stalls), zero auxiliary drift, and 100% trace equivalence.

6. Gates 4 & 5: Cross-Domain Generalization, Crossover Boundaries & Paired Statistics

6.1 Three Distinct Algebraic Operator Families (Gate 4)

To verify that $E \otimes O$ discovery is not overfit to finite-field register arithmetic, Gate 4 evaluated three structurally distinct state domains (`domain_crdt.py`, `domain_trie.py`):

1. **Domain 1 — Modular Affine Register Automata ($\mathbb{Z}^8_{4093}$)**: Conditional batch commutator on manifold $\mathcal{M} = \{\vec{R} : R_2 \equiv 0\}$ enabled by idempotent auxiliary projection $E_{\text{aff}}(R_2 \leftarrow 0)$. Reduced locks by -94.3% ($176.0 \rightarrow 10.0$) and latency by $43.1\times$ ($31{,}691\ \mu\text{s} \rightarrow 735.8\ \mu\text{s}$). *Falsification boundary*: write-hotspot contention on observable register R_1.
2. **Domain 2 — CRDT Observed-Remove Set Lattice (OR-Set)**: Speculative disjoint-partition updates (O_{crdt}) over tagged sets $(\text{Add}, \text{Tomb})$ (Shapiro et al. 2011), enabled by idempotent causal tombstone compaction E_{crdt} preserving $\mathcal{O}(\sigma) = \text{Add} \setminus \text{Tomb}$. Reduced locks by -90.0% ($200.0 \rightarrow 20.0$) and latency by $1.68\times$ ($399.1\ \mu\text{s} \rightarrow 236.9\ \mu\text{s}$), while rejecting a defective compactor that pruned live tags. *Falsification boundary*: out-of-order causal horizon delivery across partitioned replicas.
3. **Domain 3 — Transactional Key-Value Ordered Tree Store (DSW BST)**: Speculative range-partitioned updates (O_{trie}) paired with an idempotent Day–Stout–Warren (DSW) vine-to-balanced-tree retraction E_{trie} (Stout and Warren 1986) restoring $\lfloor \log_2 N \rfloor + 1$ height while preserving key–value map $\mathcal{O}(T) = \{k \mapsto v\}$.
    - **Algorithmic Crossover Negative Result**: On small in-memory workloads ($N = 150$ ops), $E_{\text{trie}} \otimes O_{\text{trie}}$ slashed lock acquisitions by -93.3% ($150.0 \rightarrow 10.0$) with 100% trace equivalence (and rejected a subtree-swapping rotator), **but increased wall-clock latency (**$232.9\ \mu\text{s} \rightarrow 396.3\ \mu\text{s}$ **in Gate 4;** $+69.6\%$ **in Gate 5)** because $\Theta(N)$ whole-tree DSW spine construction exceeds lock-avoidance savings below the large-tree ($N > 10^4$) traversal crossover threshold (Figure 1B).

6.2 Paired Non-Parametric & Parametric Statistical Validation (Gate 5)

Because baseline and candidate runtimes execute on identical sampled transaction streams, unpaired two-sample tests misstate variance. Gate 5 (`paired_statistics.py`) evaluated within-workload paired differences $D_i = X_{\text{cand},i} - X_{\text{base},i}$ across $N = 30$ **independent paired workloads per domain**, running 10,000 Monte Carlo sign-flip permutations (Good 2005), Wilcoxon signed-rank tests (Wilcoxon 1945), paired t-tests, 99% paired bootstrap confidence intervals (Efron and Tibshirani 1993), and Holm–Bonferroni family-wise error rate adjustment (Holm 1979) across all 6 endpoints (Table 4).

Table 4: Gate 5 paired statistical validation ($N = 30$ paired workloads per domain; all paired t-test $p < 10^{-12}$, all Wilcoxon $p = 1.73\times10^{-6}$, 10,000-sign-flip permutation $p = 1.00\times10^{-4}$, Holm–Bonferroni $p_{\text{adj}} = 0.0006 < 0.01$; note Domain 3 latency confirms significant $+110.97\ \mu$s rebuild overhead at $N = 150$).

Domain & Endpoint	Base Mean	Cand. Mean	Paired Diff $\bar{D}$ (99% Bootstrap CI)	Cohen's d_z	Paired t & Wilcoxon W	Holm p_{adj}
Affine: Locks	143.90	8.00	-135.90 ($[-137.57, -134.33]$)	-39.77	$t = -217.8, W = 0$	0.0006
Affine: Latency (μs)	25,797.0	1,016.1	$-24{,}780.9$ ($[-25{,}402.3, -24{,}153.8]$)	-17.86	$t = -97.8, W = 0$	0.0006
CRDT: Locks	150.00	15.00	-135.00 ($[-135.00, -135.00]$)	$-\infty$	$t = -7.4\times10^{11}$, $W = 0$	0.0006
CRDT: Latency (μs)	243.48	162.06	-81.42 ($[-88.23, -74.44]$)	-5.50	$t = -30.2, W = 0$	0.0006
KV Tree: Locks	120.00	8.00	-112.00 ($[-112.00, -112.00]$)	$-\infty$	$t = -6.1\times10^{11}$, $W = 0$	0.0006
KV Tree: Latency (μs)	159.43	270.40	$+110.97$ ($[+99.25, +124.45]$)	$+4.00$	$t = +21.9, W = 0$	0.0006

7. Autonomous Blind Discovery & Cryptographic Provenance (Gate 6)

To verify that enabler–optimizer synthesis does not rely on human-labeled pairs, `blind_discovery.py` ingested an uncurated pool of **12 unlabeled operators** (`OP_CAND_01–12`) without benchmark feedback (Table 5). By profiling precondition failure coordinates on `OP_CAND_01–03`, searching candidate retractions for idempotence ($E^2 = E$), observable preservation ($\mathcal{O} \circ E = \mathcal{O}$), and manifold containment ($\mathrm{Im}(E) \subseteq \mathcal{M}_O$), and discharging Gate 1 trace equivalence, FNWD autonomously promoted **all 3 valid compositions** (`07`⊗`01`, `08`⊗`02`, `09`⊗`03`) and rejected **all 9 defective, observable-leaking, identity, or non-idempotent operators** (`0 / 9` false promotions).

Table 5: Autonomous blind discovery triage over 12 unlabeled candidate operators (`blind_discovery.py`).

Candidate IDs	Operator Signatures	Autonomous Triage Diagnosis	Gate 1 Proof	Decision
`OP_CAND_07` ⊗ `01`	Affine Aux Reset ⊗ Affine Commutator	Idempotent on $R_2 \equiv 0$; discharges `01` defect	**PROVEN**	**PROMOTED**
`OP_CAND_08` ⊗ `02`	CRDT Tombstone Compactor ⊗ Partitioner	Purges dominated tags; preserves Add \ Tomb	**PROVEN**	**PROMOTED**
`OP_CAND_09` ⊗ `03`	DSW Tree Rebalancer ⊗ Range Partitioner	Restores $O(\log N)$ height; preserves $\{k \mapsto v\}$	**PROVEN**	**PROMOTED**
`OP_CAND_04`, `05`, `06`	Inverted Write, Payload Dropper, RAW Hazard	Unsound observable corruption / stale read	FAILED	**REJECTED**
`OP_CAND_10`	Observable-Leaking Pseudo-Enabler	Mutates observable payload R_1 ($\mathcal{O} \circ E \neq \mathcal{O}$)	FAILED	**REJECTED**
`OP_CAND_11`	Identity Map $R_{\mathrm{id}}(\sigma) = \sigma$	Zero defect projection ($\mathrm{Im}(E) \not\subseteq \mathcal{M}_O$)	FAILED	**REJECTED**
`OP_CAND_12`	Cyclic Register Permutation	Non-idempotent ($E^2 \neq E$); scrambles $\mathcal{O}(\sigma)$	FAILED	**REJECTED**

Finally, Gate 6 verified exact SHA-256 immutability across all three historical catalogs (`idempotents`: `529016c3...`, `commutators`: `01a0c025...`, `retractions`: `68f936c0...`) and all nine `nwd/atlas/` verification modules (`fnwd_v04_provenance_manifest.json`, deterministic `seed=42`).

---

8. Limitations & Conclusion

Three boundaries scope our conclusions: (1) **Algorithmic Crossover**: as proven by Domain 3's $+69.6\%$ paired latency increase at $N = 150$, algebraic soundness and -93.3% lock reduction do *not* guarantee wall-clock speedup when normalizer cost scales as $\Theta(N)$—deployment requires amortizing retraction across larger state cardinalities or amortizing batch intervals; (2) **Shared-Memory vs. WAN Asynchrony**: CRDT OR-Set trials ran under thread-level shared-memory concurrency rather than geo-distributed packet loss; and (3) **Mechanized Proof Certificates**: Gate 1 discharges finite-domain symbolic and randomized trace obligations rather than interactive Lean/Coq kernel proofs.

Within these explicit boundaries, FNWD v0.4 demonstrates that quotienting idempotent programs into a rigorous **Projection Poset** $(P, \leq)$ and pairing observational retractions with conditional commutators enables falsifiable, zero-false-promotion discovery of concurrent state-space optimizations.

Reproducibility & Standalone Package

The standalone, zero-dependency Python verification package (`nwd/atlas/`) and all three immutable JSON discovery catalogs (`nwd/results/`) reproduce all six gates and the 12-operator blind discovery audit in < 5 seconds via `python -m nwd.atlas.run_atlas_v04`.

References

Clements, Austin T., M. Frans Kaashoek, Nickolai Zeldovich, Robert T. Morris, and Eddie Kohler. 2015. "The Scalable Commutativity Rule: Designing Scalable Software for Multicore Processors." *ACM Transactions on Computer Systems* 32 (4): 10:1–47. https://doi.org/10.1145/2699681.

Efron, Bradley, and Robert J. Tibshirani. 1993. *An Introduction to the Bootstrap*. Chapman & Hall/CRC.

Eswaran, Kapali P., Jim N. Gray, Raymond A. Lorie, and Irving L. Traiger. 1976. "The Notions of Consistency and Predicate Locks in a Database System." *Communications of the ACM* 19 (11): 624–33. https://doi.org/10.1145/360363.360369.

Good, Phillip. 2005. *Permutation, Parametric, and Bootstrap Tests of Hypotheses*. 3rd ed. Springer.

Herlihy, Maurice P., and Jeannette M. Wing. 1990. "Linearizability: A Correctness Condition for Concurrent Objects." *ACM Transactions on Programming Languages and Systems* 12 (3): 463–92. https://doi.org/10.1145/78969.78972.

Holm, Sture. 1979. "A Simple Sequentially Rejective Multiple Test Procedure." *Scandinavian Journal of Statistics* 6 (2): 65–70.

Howie, John M. 1995. *Fundamentals of Semigroup Theory*. Vol. 12. London Mathematical Society Monographs. Oxford University Press.

Lowe, Gavin. 2017. "Testing for Linearizability." *Concurrency and Computation: Practice and Experience* 29 (4): e3928. https://doi.org/10.1002/cpe.3928.

Rinard, Martin C., and Pedro C. Diniz. 1997. "Commutativity Analysis: A New Analysis Technique for Parallelizing Compilers." *ACM Transactions on Programming Languages and Systems* 19 (6): 942–91. https://doi.org/10.1145/267959.269969.

Scott, Dana S. 1976. "Data Types as Lattices." *SIAM Journal on Computing* 5 (3): 522–87. https://doi.org/10.1137/0205037.

Shapiro, Marc, Nuno Preguiça, Carlos Baquero, and Marek Zawirski. 2011. "Conflict-Free Replicated Data Types." *Stabilization, Safety, and Security of Distributed Systems (SSS 2011)*, Lecture notes in computer science, vol. 6976: 386–400. https://doi.org/10.1007/978-3-642-24550-3_29.

Stout, Quentin F., and Bette L. Warren. 1986. "Tree Rebalancing in Optimal Time and Space." *Communications of the ACM* 29 (9): 902–8. https://doi.org/10.1145/6592.6599.

Wilcoxon, Frank. 1945. "Individual Comparisons by Ranking Methods." *Biometrics Bulletin* 1 (6): 80–83. https://doi.org/10.2307/3001968.